

# Weekly report (2013.5.27~6.2)

## Done

- 1) This week, I concentrate on the optimization of the front-end. As shown in Figure 1.  
(a) is the previous version, requests from browser are received by php, then php calls a c++ render to do the computing. In this way, the render need to read textures every time it's brought up by php, and frequent start and destroy of c++/php instances also brings overhead.

Now, it's replaced by a multi-threaded fastcgi, it's a service written in c++ and permanent in the memory, Requests are delivered to threads when received, and it loads all the textures into memory when brought up (only once), so it's more efficient compared to the previous one.

And I reconstructed the code by the way.

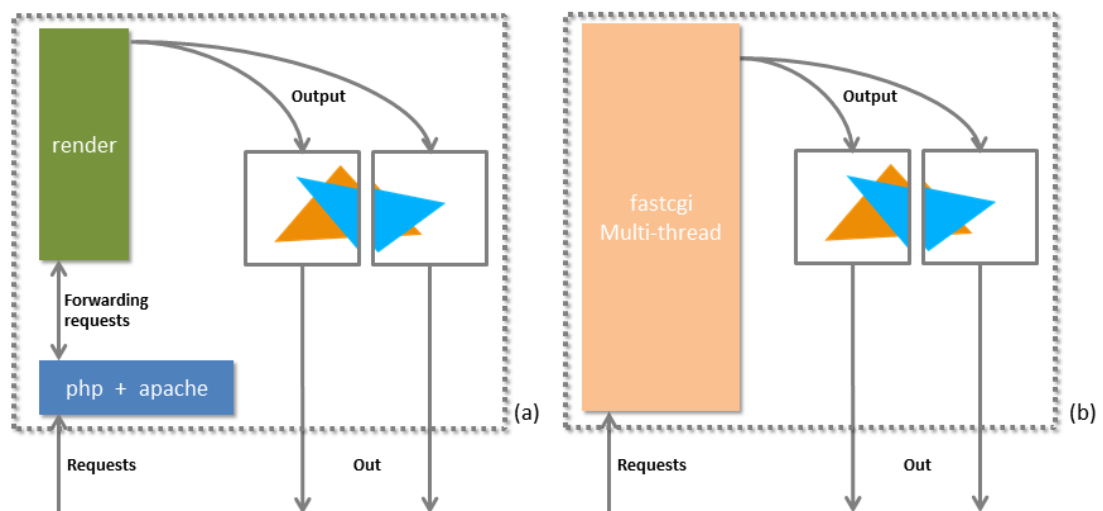


Figure 1 optimization based on fastcgi

- 2) Spend some time implement the MPI communicator to replace the local one in the parallel framework designed by sikan.
- 3)

Figure 2 schedule

Mon.	Tue.	Wed.	Thu.	Fri.
aliyun	308			

## To Do

- 1) Compare my work with previous papers.
  - a) Cloud computing: Hu.
  - b) layered depth image.